WEARABLE GESTURE RECOGNITION WITH

HETEROGENEOUS CAMERAS

A Thesis
Presented to
The Academic Faculty

by

Tyler LaBean

In Partial Fulfillment
of the Requirements for the Degree
Bachelor of Science in Computer Science
with the Research Option in the
School of Computer Science

Georgia Institute of Technology
August 2016

WEARABLE GESTURE RECOGNITION WITH HETEROGENEOUS CAMERAS

Approved by:

Dr. Thad Starner, Advisor
School of Computer Science
Georgia Institute of Technology

Dr. James Hayes
School of Computer Science
Georgia Institute of Technology

Peter Presti
Interactive Media and Technology Center
Georgia Institute of Technology

Date Approved:

TABLE OF CONTENTS

ABSTRACT

The purpose of this research was to create a wearable system that recognizes gestures of the user, allowing interaction through hand gestures. The user wears a hat mounted with a regular optical camera and a thermal camera. The combination of these two heterogeneous video streams was used to recognize the user's gestures in many conditions and environments. First, corners were detected from contrast stretched images using the Shi-Tomasi method. The movement of these corners was then tracked using Lucas-Kanade optical flow analysis. Groups of corners that moved together were defined using hierarchical cluster linkage analysis. To determine how these groups moved with time, a connected components analysis was employed. The motion path was reduced into its cardinal and semi cardinal vector components to encode the motion vector. Subsequently, this data was used to train hidden Markov models for each gesture and each camera. After the evaluation of gesture priority over all hidden Markov models, principal components analysis was performed on this gesture prioritized set to train a one vs one Multiclass recognizer. Finally, a confusion matrix was generated indicating a recognition success rate of 87%. An analysis was performed on the robustness of the algorithm under various luminance, heat and image variance conditions. The contribution of combining optical and thermal video streams vs utilizing either as a single video stream input and found to be a great advantage. Additionally, a video database of gestures was created and will be released so that other researchers can compare algorithms and benchmarks using the same data-set.

CHAPTER 1

INTRODUCTION

Gesture recognition for wearable devices is a field of study that has become more important as mobile and ubiquitous computing rises in popularity. As our devices become more powerful, smaller, and more socially acceptable, so too must our methods of interacting with these devices. A full sized physical mouse and keyboard setup is not viable when the entire computer is built into your wristwatch. Mobile computing is becoming ubiquitous, yet interaction methods have not changed significantly since the advent of the touchscreen. More advanced and natural methods of communicating with mobile computers is necessary for wearable computing to take off, just as the touchscreen was a large factor in the success of smart-phones and tablets.

There has been much research into various methods of Human Computer Interaction (HCI) using computer vision to recognize human gestures [1]. Specifically the recognition of hand gestures is important because as Hasan and Abdul-Kareem state, "It is a natural medium for communication between humans and thus the most suitable tool for HCI" [1].

There are different ways to accomplish the task of gesture recognition. If the target application can accommodate wearable technology the possibilities include accelerometer and other wearable sensors or visual techniques. However, some applications do not allow for sensors such as accelerometers to be attached to the user [2]. In this case a visual method must be employed.

When creating any vision-based gesture recognition system, designers must choose between active or passive techniques [3]. An active technique involves projecting light onto a scene then using information about that projected light to gather information about the environment. Sometimes laser light is bounced onto objects to determine depth or patterned Infrared (IR) light is projected onto a scene and the distortions and shadows in this projection provide shape information. Passive techniques do not project anything onto the scene and instead use multiple cameras, shading, or silhouettes to determine shape and depth information. According to Kress and Lee, active methods can often produce better results, but at a greatly increased power consumption (due to the energy needed to project light onto the scene) [3]. Because of this limitation, when designing for wearable or mobile applications where battery consumption can be an important factor, passive methods may be favored due to their energy efficiency.

Traditionally passive visual based gesture recognition systems have used only one camera. Algorithms have been developed to use color cameras, depth cameras, and thermal cameras separately [4]. There are strengths and weaknesses for each of these types of cameras depending on background conditions. However, none of these camera types are adequate in all environments. Variations in lighting, temperature, and background motion reduce the ability of individual video-streams to segment and recognize gestures [4].

One way to create a passive vision-based gesture recognition system is with multiple heterogeneous cameras. There has been some research into using heterogeneous cameras for gesture recognition [5]. Specifically thermal cameras are chosen in addition to optical cameras because thermal data often is very good at segmenting an image into "Skin" and "Not Skin" [5]. If no thermal camera is present, skin-color segmentation is used to determine the position of the hand in an image. However some lighting or background conditions can cause this method to fail [5]. In these situations the addition of a thermal camera can greatly improve results as there is commonly a clear overlap of skin-color in the optical domain and skin-temperature in the thermal domain. Objects in the background that are roughly skin-color can be ignored if they are not roughly skin-temperature, and vice versa.

Much of the research into thermal-optical image segmentation for gesture recognition used a third person camera perspective [6]. In this mode cameras are fixed to the environment and are pointed at a subject who then uses hand gestures to interact with a stationary computer. Additionally, much of the testing is done inside a lab with tightly controlled parameters such as ambient light and background [7]. The current study uses images from a first person perspective, meaning that the cameras are attached to the user and look outwards. Also, varied backgrounds and lighting were tested and recorded.

The most common way to recognize gestures is to start by segmenting the image into different regions of interest. Many methods use skin-color segmentation or background subtraction to find the specific object or objects that will perform the gesture [8]. Alternatively, depth cameras find objects that are close in the foreground and use them as the gesturing objects [4]. Previous algorithms have used dense optical flow to segment the images into motion blobs which are then tracked over time [9]. My approach was to first search for strong corners to track using the Shi-Tomasi method [10]. I chose to use the Lucas-Kanade method of sparse optical flow to track the motion of these corners over time [11].

There are various options for grouping the strong corners that moved together throughout the video-streams into objects. One option is to use K-Means clustering to find the objects, but this requires the number of mobile objects in the scene to be predetermined. Additionally, K-Means clustering has a bias toward creating clusters of equal size, which is not ideal if the target object may be significantly smaller in size than the background or other slower moving objects. Therefore, I chose to use a linkage agglomerative clustering so that scenes with variable numbers of moving objects can be determined in a manner that does not depend of the size of those objects [12]. Thus the moving objects are segmented from the background and from each other.

When attempting to recognize a complex signal, Dynamic Time Warping can be employed [13]. However, when attempting to recognize human gestures, Hidden Markov Models are often used due to variation between samples [8].

Hidden Markov Models are used by training an HMM to recognize one particular pattern. In the domain of gesture recognition, an HMM is trained to specificity recognize one gesture. Then when testing a new gesture, each HMM is tested individually and often the HMM that outputs the highest likelihood response for that gesture is selected in a winner take all fashion [4]. Instead, I chose to use a One-Vs.-One multiclass classification schema for interpreting the output of a set of HMMs for a particular test gesture.

This thesis outlines a vision-based gesture recognition system that uses two heterogeneous video-streams to create a view-based motion segmentation system. The results of this study demonstrate the advantages of the combination of disparate input streams when attempting to recognize gestures in highly variate environments. An important contribution of this study is the creation of an annotated dataset of simultaneous color and thermal video-streams for further research into the field of gesture recognition.

CHAPTER 2

METHODS

Figure 1: Data Collection Rig

The device is a Raspberry Pi 2 with the stock Raspberry Pi camera module and the FLIR Lepton thermal camera with the Raspberry Pi mounting board. The cameras are mounted side by side to the brim of a hat and point downwards (Figure 1). The data collection software is coded in Python and makes use of the Open-CV python library (CV2) and the machine learning library ScikitLearn. Data from the Raspberry Pi is offloaded to a desktop computer for analysis. A database of video-streams of annotated gestures was created using the data collection device. In these video-streams, the user gestures by moving a hand along a path which is then tracked by the cameras. Specifically, two hundred samples of combined visual and thermal recordings of hand gestures of me drawing the numbers 1,2,3,4 and 5 in space with my index finger were recorded as follows: A single user (me) walked to a location, stopped moving, started recording, moved my hand up to approximately the same height and distance away from the cameras each time, performed the gesture for number 1, stopped the recording and in the same location repeated the gestures for the other four numbers and stopped recording. The gestures representing the numbers were similar in size and as stereotyped as possible such that only the background would vary. Backgrounds were randomly chosen both in indoor and outdoor settings and included a wide range of ambient temperatures and light levels. Ultimately nearly 40 examples of each gesture was recoreded.

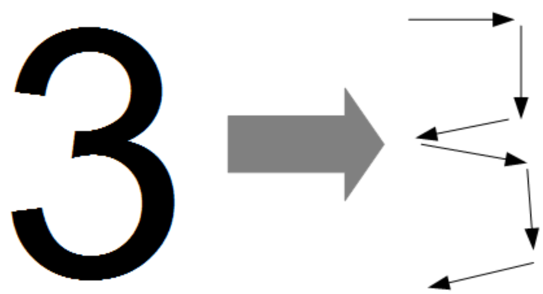


Figure 2: Movement Vectors of the Symbol '3'

These videos are annotated by hand, recording what types of gestures were performed and when the gestures occurred. Figure 2 is an example of how a gesture for the numeral three might be represented as movement vectors. The hand sweeps out the shape of the symbol, which is a series of movements.

GESTURE RECOGNITION ALGORITHM

Contrast Stretch Images

Perform Shi-Tomasi Corner Detection

Perform Lucas-Kanade Optical Flow Analysis

Generate Motion Groups Using Hierarchical Cluster Linkage

Generate Weighted Edge List Of Motion Groups

Prune Edges With Low Weight And Generate Unweighted Edge List Of Motion Groups

Perform Connected Components Analysis Over Time

Sort Connected Components By Path Length

Decompose Path Into Cardinal And Semi-cardinal Directions And Threshold To Encode Motion Vector

Filter out initial upward motion of hand

Train Hidden Markov Model For Each Gesture And Each Camera

Evaluate Gesture Priority Over All Hidden Markov Models

Use Principal Component Analysis on Gesture Priority to Train One-Vs.-One Multiclass Recognizer

Figure 3A: Raw Optical Image

Figure 3B: Contrast Stretching

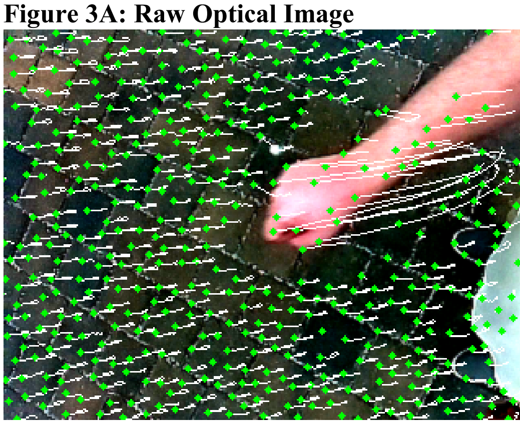


Figure 3C: Optical Flow Analysis

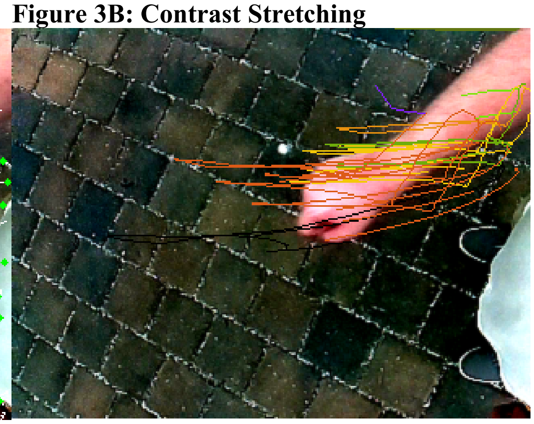


Figure 3D: Movement Grouping

A set of features was extracted from the video data and used to create a set of Hidden Markov Models as follows. The hand motion per frame is segmented from the background through a series of algorithms starting with the raw images (Figure 3A). First the contrast of the image is stretched (Figure 3B). Then the Lucas-Kanade method of optical flow is applied to the corners found via the Shi-Tomasi method (Figure 3C) [10, 11]. For the Shi-Tomasi method a block size of 7 (Color) and 5 (Thermal) was used. The optimal search window for the Lucas-Kanade method was 11x11 pixels for both image types. This produces discrete 'tracks' where discernible corners moved throughout the video. Next, these tracks are grouped via an agglomerative clustering algorithm using the the movement of the tracks on a per-frame basis [12]. Then the track linkages are collapsed into clusters based on the euclidean difference with an inconsistency threshold of 1.15. Specifically, a rolling average of the motion of the track minus the average motion of other tracks is used. This was done because the system is head-mounted, and it is possible that the camera position relative to the scene will shift. Thus the movement of objects relative to other objects forms the basis for segmentation; corners that move together are grouped together. The track groupings are recorded into an edge list of an undirected graph. The graph represents the connections between tracks. The edge weights represent how strongly correlated the movement of one track was compared to another track. This edge list is pruned based on connection strength such that only tracks that have significant overlap and simultaneous correlation remain. This thresholded edge list is used in a connected components algorithm to determine which tracks are correlated enough to be considered discrete objects in the scene.

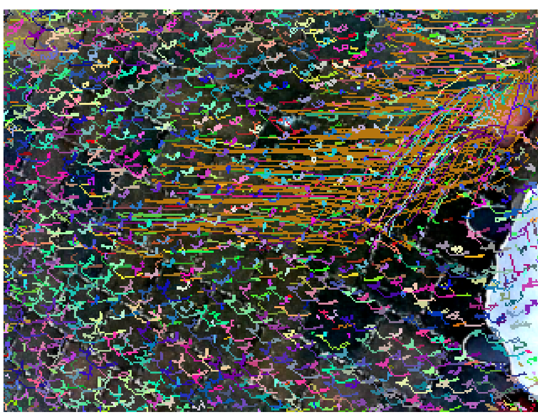


Figure 4A: Connected Components

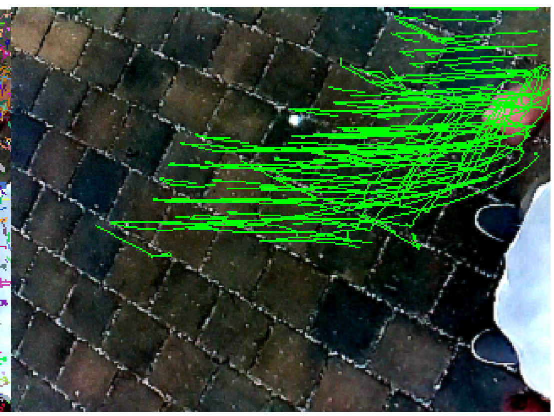


Figure 4B: Object with Longest Path Length

Using the object list created through the connected component algorithm, along with the overall path length of said objects found by summing the average of the track movement from one frame to the next, the objects can be sorted by how long they persisted and how much they moved in the image over time. Because the cameras point downwards at a steep angle, there are rarely any moving objects that persist in the video other than the background and the hand which is forming the gesture. Additionally, the total movement of the background is always significantly less than the total movement of the hand while gesturing. Thus the single object that moved the most over time can be assumed to be the hand which is gesturing.

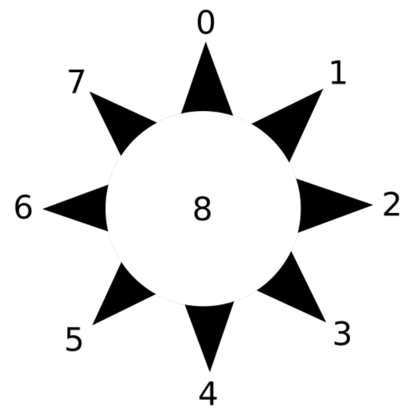


Figure 5: Movement Direction Codes

Next, the direction of the hand movement per frame is reduced into the cardinal and semi-cardinal directions and then mapped into the alphabet 0-7 as shown in Figure 5. If the speed of motion is below a threshold, the code returned is 8 instead. The movement threshold was found experimentally.

Motion Codes

Raw Movement Direction Codes	8, 8, 8, 8, 6, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 0, 0, 0, 0, 0, 8, 8, 8, 8, 8, 4, 3, 4, 4, 4, 4, 4, 4, 4, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 2, 2, 2, 2, 2, 8, 8, 8, 2, 8, 2, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8
Remove Slow Motions	6, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 0, 0, 0, 0, 0, 4, 3, 4, 4, 4, 4, 4, 4, 4, 2, 2, 2, 2, 2, 2, 2
Remove Intro and Outro Motions	4, 3, 4, 4, 4, 4, 4, 4, 4, 2, 2, 2, 2, 2, 2, 2

The movement codes are then pruned to remove the introductory movement upwards that nearly all of the gestures share. Because in the data-set created for this study the gestures are performed starting from a resting position with the hands at the side of the body, this upwards movement is present in almost all of the gestures. Thus before each gesture truly starts, the hand must move upwards and into position in front of the torso. If there is not a clear upwards segment early on in the motion code list, then the gesture is is not pruned. Similarly to the upwards introduction motion, there is a consistent downwards motion after a gesture is completed where the hand returns to a resting pose. This 'outro' motion is pruned in the same way as the 'intro' motion.

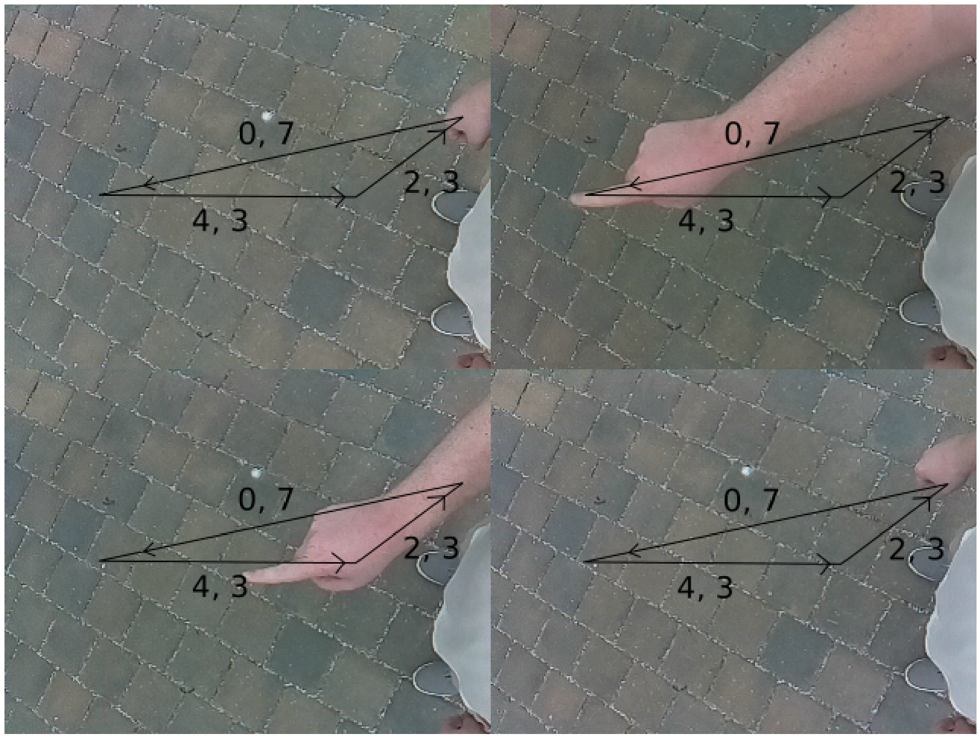


Figure 6: Motion Codes for '1' Symbol

The above images (Figures 3, 4, and 6) and resulting motion codes table describe one example of the gesture for the symbol '1'.

In three steps:

-The hand moves upwards and into position: Motion Code 0, 7

-The hand moves downwards in the actual gesture: Motion Code 4, 3

-The hand moves back to the resting position: Motion Code 2, 3

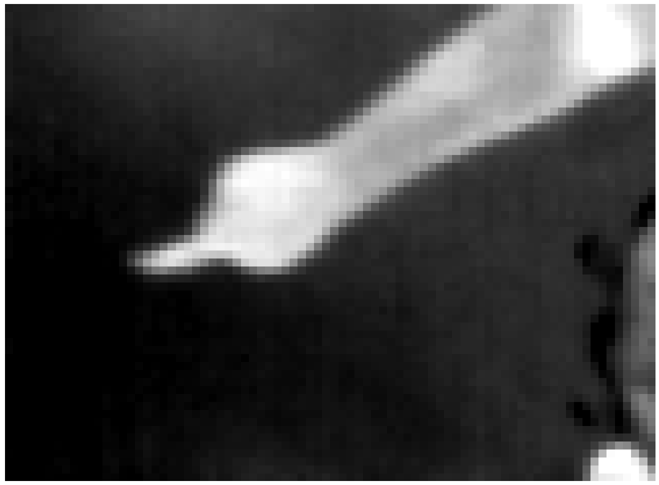


Figure 7A: Raw Thermal Image

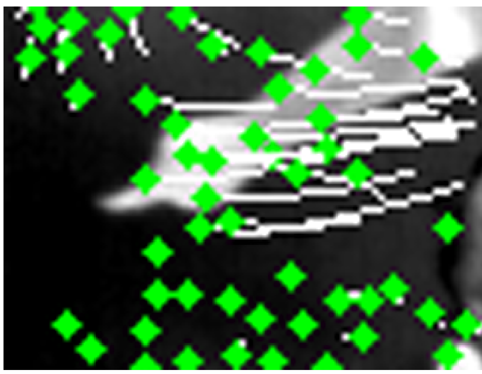


Figure 7B: Optical Flow Analysis

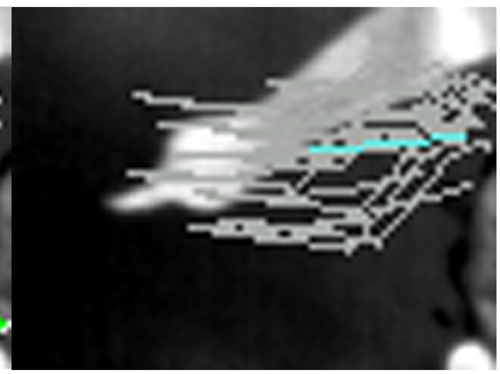


Figure 7C: Movement Grouping

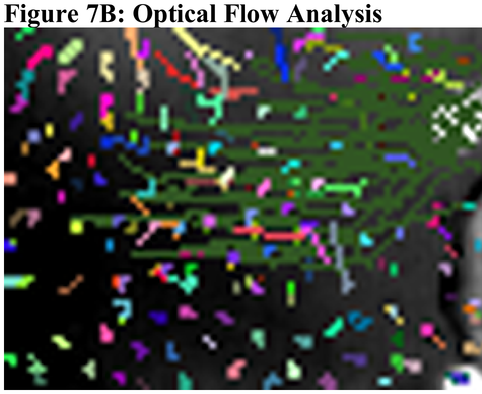


Figure 7D: Connected Components

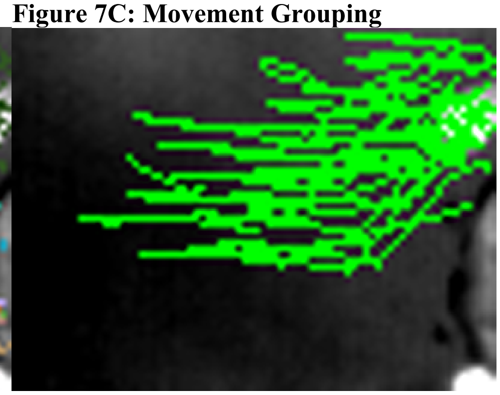


Figure 7E: Object with Longest Path

The same pipeline is used on the thermal video stream to create a separate series of motion codes. Thus there is a series of motion codes for both the optical and thermal video streams.

As is visible in the data, there are discrete hidden states (what part of the gesture is currently being performed) and emitted signals (motion codes) that have different emission probabilities based on the hidden state. A Hidden Markov Model (HMM) is trained for each image type per gesture. Thus there is a set of 10 HMMs (5 gestures and 2 image types). Then using the same training data used to create the HMMs, a One-Vs-One ensemble classifier is trained to use the log probabilities returned by the 10 HMMs to ultimately classify the gesture. Internally the One-Vs-One classifier uses a set of quadratic discriminant classifiers that return the likelihood that a given gesture is the target gesture over another gesture.

The trained classifier as well as the annotated video database will be distributed for use in other research.

CHAPTER 3

RESULTS

With optimal parameters and 10-Fold Cross Validation, the classifier reached an average recognition rate of 86.98% when using the thermal and optical images together. The confusion matrix of the classifier shows that the gestures '2' and '3' were confused with each other more than any other pair of gestures.

Optical + Thermal Confusion Matrix

Actual\Predicted	'1'	'2'	'3'	'4'	'5'
'1'	100.00%	0.00%	0.00%	0.00%	0.00%
'2'	5.00%	75.00%	17.50%	2.50%	0.00%
'3'	0.00%	18.42%	76.32%	0.00%	5.26%
'4'	2.70%	0.00%	0.00%	97.30%	0.00%
'5'	0.00%	5.41%	8.11%	0.00%	86.49%

If a new classifier is trained and tested using only the optical data, the results are lower, with an average recognition rate of 76.04% after 10-Fold Cross Validation.

Optical Only Confusion Matrix

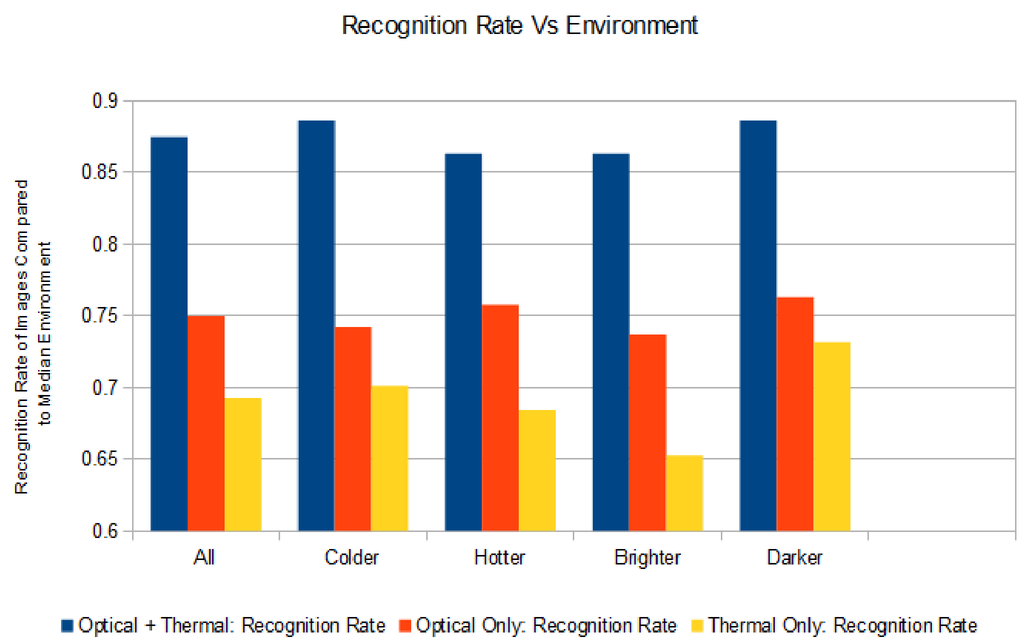
Actual\Predicted	'1'	'2'	'3'	'4'	'5'
'1'	100.00%	0.00%	0.00%	0.00%	0.00%
'2'	7.50%	45.00%	45.00%	0.00%	2.50%
'3'	5.26%	28.95%	63.16%	0.00%	2.63%
'4'	10.81%	0.00%	0.00%	89.19%	0.00%
'5'	2.70%	8.11%	5.41%	0.00%	83.79%

Additionally if a classifier is trained and tested using only the thermal data the results are also significantly worse than the combination optical-thermal data with an overall recognition rate of 68.75%

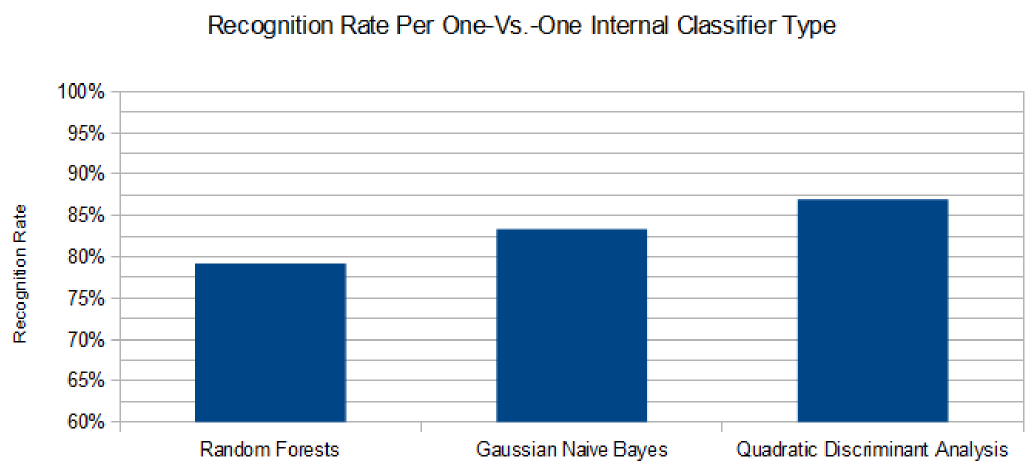
Thermal Only Confusion Matrix

Actual\Predicted	'1'	'2'	'3'	'4'	'5'
'1'	97.50%	0.00%	0.00%	2.50%	0.00%
'2'	17.50%	30.00%	50.00%	2.50%	0.00%
'3'	2.63%	26.32%	63.16%	2.63%	5.26%
'4'	18.92%	0.00%	0.00%	81.08%	0.00%
'5'	0.00%	8.11%	13.51%	5.41%	72.97%

When the results are broken into groups based on environmental conditions the variance in recognition rate compared to environmental factors can be assessed. The gesture recognizer was trained as before, but during testing the test gesture's average temperature and brightness compared to the median training datum in temperature and brightness was recorded and the results plotted.



Additionally the choice of internal classifier for the One-Vs-One classifier had an effect on the recognition rate.



The ability for the system to use the priority vector produced by the 10 HMMs was affected by the choice of internal classification. In this study a quadratic discriminant analysis produced the best results.

CHAPTER 4

DISCUSSION

The creation of a database of gestures created in stereotyped fashion but under different conditions of background luminance, heat and image variance is of value to the machine learning community. Sharing this database was a major goal of my research project particularly since datasets with both thermal and visual images from parallel vantage is lacking in the literature.

In the future this algorithm could be tested using either a HMM using Gaussian emissions instead of discrete emissions. Some of the difficulty of categorizing gestures in this study was due to over-reliance on specific angles of motion. At times a motion that from the user's perspective was straight forwards away from the body, which should be quantized as a motion code of '0', could be seen as one of the bordering codes, '1' or '7'. This effect seemed to last for an entire gesture, thus all instances of that vertical motion code is one step clockwise or anti-clockwise. This shifting effect is most clearly seen in gestures such as the symbol '4' as it is comprised of mostly straight line motions that are highly affected by a one step clockwise or anti-clockwise shift. This effect may be caused by slight angle changes in head position or even changes in the exact positioning of the cameras on the head. If a HMM with Gaussian emissions was used, the most relevant movements could be grouped algorithmically, instead of arbitrarily by the cardinal and semi-cardinal directions. Additionally, a Gaussian emission based system may be able to recognize the speed of motion as well as direction and use that to better distinguish irrelevant motions and the movements that actually comprised the gesture. Currently, even when the hand is correctly selected and the path of the hand extracted, is is difficult to determine exactly which motions are part of any lead-up and follow-up to the gesture. Because the dataset was created with a push-to-record based system, there is always a predictable movement upwards from resting position to the gesture's starting position. This motion can be cut away from the motion codes before training the HMMs; however, some of the gesture could accidentally be removed. It is possible that the confusion between the symbol '2' and '3' is due to this.

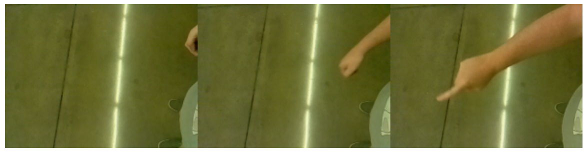


Figure 8A: Symbol 2 Upwards Motion

Figure 8B: Symbol 2 Curved-Sweep Motion

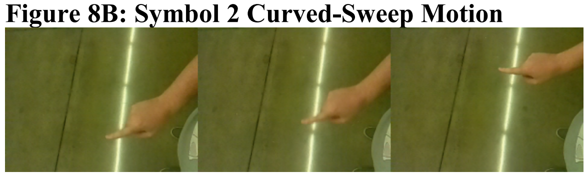


Figure 8C: Symbol 2 Horizontal Straight Motion

Figure 8D: Symbol 2 Return Motion

Above is an example of the gesture for the symbol '2' (Figure 8). A significant portion of the video is an initial upwards movement into position (Figure 8A). Then, the first part of the actual gesture is the inwards sweep of the top of the '2' (Figure 8B). Next, the horizontal movement of the base of the '2' (Figure 8C). Finally, the hand returns to the resting position (Figure 8D). Below is an example of the gesture for the symbol '3' (Figure 9). Again a large part of the video is the initial upwards movement (Figure 9A). Then the downwards arc of the top of the '3' (Figure 9B). Next, the second downwards arc of the '3' (Figure 9C). Then the hand drops to the resting pose (Figure 9D).

Figure 9A: Symbol 3 Upwards Motion

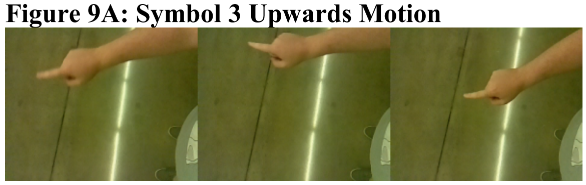


Figure 9B: Symbol 3 First Downwards Arc

Figure 9C: Symbol 3 Second Downwards Arc

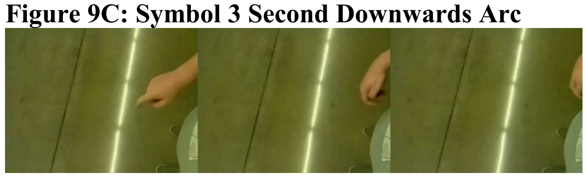


Figure 9D: Symbol 3 Return Motion

If the return motion of the '3' gesture is pruned too much, some of the second downwards arc (Figure 9C) may be lost. In this event the remaining movement vector is extremely similar to the movement codes generated for the symbol '2'. I believe this is the cause of the low recognition rate between '2' and '3' relative to the other pairs of gestures.

Other studies have found the same issues with a push-to-record based data collection system [14]. In these cases some groups have simply changed the types of gestures used in order to reduce the likelihood that the actual gesture will be confused with the lead-up and follow-through of a gesture [14].

CHAPTER 5

CONCLUSION

Ultimately, the method described in this thesis was more invariant to environmental conditions than similar methods. The combination of color and thermal video-streams was able to maintain a consistent recognition rate regardless of ambient light levels and average temperature. This method did not rely on skin-color segmentation which is vulnerable to variations in lighting [15]. It also had little variance in recognition rate due to changes in temperature. The primary benefit of this method was in its robustness. If a vision-based gesture recognition system is required to function in a wide range of conditions, this study shows that the combination of thermal and color images is a viable approach.

REFERENCES

1 Hasan, H., and Abdul-Kareem, S.: 'Human-computer interaction using vision-based hand gesture recognition systems: a survey', Neural Computing & Applications, 2014, 25, (2), pp. 251-261

2 Evett, L., Burton, A., Battersby, S., Brown, D., Sherkat, N., Ford, G., Liu, H., and Standen, P.: 'Dual camera motion capture for serious games in stroke rehabilitation', in Editor (Ed.)^(Eds.): 'Book Dual camera motion capture for serious games in stroke rehabilitation' (2011, edn.), pp. 1-4

3 Kress, B., and Lee, J.: 'Optical gesture sensing and depth mapping technologies for head-mounted displays: an overview', Photonic Applications for Aerospace, Commercial, and Harsh Environments Iv, 2013, 8720, pp. 13

4 Appenrodt, J., Al-Hamadi, A., and Michaelis, B.: 'Data gathering for gesture recognition systems based on single color-, stereo color-and thermal cameras', International Journal of Signal Processing, Image Processing and Pattern Recognition, 2010, 3, (1), pp. 37-50

5 Zeng, B., Wang, G., and Lin, X.: 'A hand gesture based interactive presentation system utilizing heterogeneous cameras', Tsinghua Science and Technology, 2012, 17, (3), pp. 329-336

6 Rautaray, S.S., and Agrawal, A.: 'Vision based hand gesture recognition for human computer interaction: a survey', Artificial Intelligence Review, 2015, 43, (1), pp. 1-54

7 Pisharady, P.K., and Saerbeck, M.: 'Recent methods and databases in vision-based hand gesture recognition: A review', Computer Vision and Image Understanding, 2015, 141, pp. 152-165

8 Hyeon-Kyu, L., and Kim, J.H.: 'An HMM-based threshold model approach for gesture recognition', IEEE Transactions on Pattern Analysis and Machine Intelligence, 1999, 21, (10), pp. 961-973

9 Cutler, R., and Turk, M.: 'View-based interpretation of real-time optical flow for gesture recognition', in Editor (Ed.)^(Eds.): 'Book View-based interpretation of real-time optical flow for gesture recognition' (1998, edn.), pp. 416-421

10 Jianbo, S., and Tomasi, C.: 'Good features to track', in Editor (Ed.)^(Eds.): 'Book Good features to track' (1994, edn.), pp. 593-600

11 Lucas, B.D., and Kanade, T.: 'An iterative image registration technique with an application to stereo vision'. Proc. Proceedings of the 7th international joint conference on Artificial intelligence - Volume 2, Vancouver, BC, Canada1981 pp. Pages

12 Ross, G.J.S.: 'Algorithm AS 15: Single Linkage Cluster Analysis', Journal of the Royal Statistical Society. Series C (Applied Statistics), 1969, 18, (1), pp. 106-110

13 'Dynamic Time Warping': 'Information Retrieval for Music and Motion' (Springer Berlin Heidelberg, 2007), pp. 69-84

14 Kratz, S., and Back, M.: 'Towards Accurate Automatic Segmentation of IMU-Tracked Motion Gestures'. Proc. Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems, Seoul, Republic of Korea2015 pp. Pages

15 Moeslund, T.B., and Granum, E.: 'A Survey of Computer Vision-Based Human Motion Capture', Computer Vision and Image Understanding, 2001, 81, (3), pp. 231-268